

Article

An Inexpensive Method for Kinematic Calibration of a Parallel Robot by Using One Hand-Held Camera as Main Sensor

Alberto Traslosheros ¹, José María Sebastián ^{2,*}, Jesús Torrijos ², Ricardo Carelli ³ and Eduardo Castillo ⁴

¹ Postgrado, Universidad Aeronáutica de Querétaro, Carretera Estatal 200, Tequisquiapan 22154, 76270 Colón, Mexico; E-Mail: altrami@hotmail.com

² CAR-UPM-CSIC, Universidad Politécnica de Madrid, c/José Gutiérrez Abascal 2, 28006 Madrid, Spain; E-Mail: jesus.torrijos.ramirez@alumnos.upm.es

³ Instituto de Automática, Universidad Nacional de San Juan, Av. San Martín Oeste 1109, 5400 San Juan, Argentina; E-Mail: rcarelli@inaut.unsj.edu.ar

⁴ Instituto Politécnico Nacional, CICATA, México, Department of Mechatronics, Cerro Blanco 141, 76090 Querétaro, Mexico; E-Mail: ecastilloca@ipn.mx

* Author to whom correspondence should be addressed; E-Mail: jsebas@etsii.upm.es; Tel.: +34-91-3363-061; Fax: +34-91-3363-010.

Received: 21 May 2013; in revised form: 27 June 2013 / Accepted: 27 July 2013 /

Published: 5 August 2013

Abstract: This paper presents a novel method for the calibration of a parallel robot, which allows a more accurate configuration instead of a configuration based on nominal parameters. It is used, as the main sensor with one camera installed in the robot hand that determines the relative position of the robot with respect to a spherical object fixed in the working area of the robot. The positions of the end effector are related to the incremental positions of resolvers of the robot motors. A kinematic model of the robot is used to find a new group of parameters, which minimizes errors in the kinematic equations. Additionally, properties of the spherical object and intrinsic camera parameters are utilized to model the projection of the object in the image and thereby improve spatial measurements. Finally, several working tests, static and tracking tests are executed in order to verify how the robotic system behaviour improves by using calibrated parameters against nominal parameters. In order to emphasize that, this proposed new method uses neither external nor expensive sensor. That is why new robots are useful in teaching and research activities.

Keywords: kinematic calibration; external sensor calibration; one-camera-in-hand sensor; parallel mechanism; parallel robot; visual servoing

1. Introduction

Parallel robots or parallel kinematic machines (PKM), in contrast to serial robots, are characterized by high structural stiffness, high load operation, high speed and acceleration of the end effector and high accuracy positioning of the end effector. This is why parallel robots are commonly used in tasks such as surgery, material machining, electronics manufacturing, and product assembly among many other applications. In order to reach such high robot accuracy, it is necessary to avail robust and reliable calibration methods, which is quite difficult to obtain from both theoretical and practical point of view, even if it could be performed off-line. Robot accuracy can be affected by increasing backlash due to robot operation, thermal effects, robot control, robot dynamics [1], manufacturing errors and element deformation [2,3]. Nowadays, there are many robot calibration methods, so it is not easy to provide a good classification in order to know the advantages and disadvantages of each one. Furthermore, there are even some hybrid calibration methods. However, in general, it is possible to classify different calibration strategies in three main groups by considering the location of the measurement instruments and their additional elements (for a good overview, see [4]). Thus, there are external, constrained and auto calibration methods. *External calibration* is based on measurements of several poses (if orientation is considered) of the robot end effector (or other structural element) by using an external instrument. *Constrained calibration* methods rely on mechanical elements to constraint some kind of motion of the robot during the calibration process. This method is generally simple and it is considered the most inexpensive. And finally, *auto calibration* methods are those that calibrate the robot automatically, even during the robot operation [2,5–7]. In general auto (or self) calibration methods are more expensive due to their complexity. They are even used for including redundant sensors [8], redundant elements or more complex algorithms. The calibration method shown in this article is considered an external calibration method.

As previously mentioned, external calibration can generally be carried out by measuring pose parameters of the end effector of the robot (or other elements of the robot) completely or partially. Measurements of the pose of a platform can be made with a laser [9] and a coordinate measuring machine (CMM) [2,10], commercial visual systems [11], laser sensor [12], including additional passive legs [13], interferometers [14], LVDT and inclinometers in [15,16], theodolite [17], gauges [18], double ball bar (DBB) [19], inspection of a machine part that is dedicated to the calibration process [2,20], accelerometers [21], or visual systems and patterns that have been widely studied (*i.e.*, chess-board) [6,22–25]. The above examples represent different strategies that can be used in order to obtain kinematic information about the robot, but in general, calibration methods impose virtual or real constraints on the poses of the end effector (or mobile elements). By choosing an appropriate method, a calibration can be an economical and practical technique in order to improve accuracy of a parallel robot (PKM) [25].

Generally, the goal of a kinematic calibration process is to obtain the constraints of quite large measured poses (called calibration poses) in order to determine which geometry (distances or angles of elements) of the robot is the best to satisfy the above constraints [26]. It should be mentioned that there are more recent approaches, which can identify both model structure and parameters [27], obtain a range of solutions (interval analysis) that guarantee parameter values to satisfy the calibration equations or approaches that obtain a few, but high accuracy poses [9]. The above basic idea has been applied to the calibration of serial and parallel robots. The main differences are found in the measurement instruments and strategies. Visual methods are becoming more popular due to their simplicity and low cost compared to other methods. Parallel robots visual techniques were first proposed by Amirat [28]. Stereo visual methods are developed [29,30] but other visual methods based on a monocular system [6,22–25,31,32] propose using a pattern or marks on a flat surface. These kinds of patterns are employed in camera calibration where intrinsic and extrinsic parameters [33] can be obtained. By means of the extrinsic parameters of the camera it is possible (by attaching the camera or the pattern to the end effector of the robot) to obtain the pose of the end effector of the robot and consequently consider the poses as calibration poses.

In this paper a novel, inexpensive, and external calibration method to calibrate a parallel robot with three DOF is proposed. In order to obtain the joint and 3D incremental positions, the proposed method utilizes positions (not absolute but incremental positions) from the resolvers of the motors and visual information obtained from a spherical element. Once the above information is obtained, it is used in the constraint equations to solve them numerically and obtain the best set of geometrical parameters that will satisfy the set of equations. Finally, several tests are executed in order to check the best behaviour with calibrated parameters against nominal parameters.

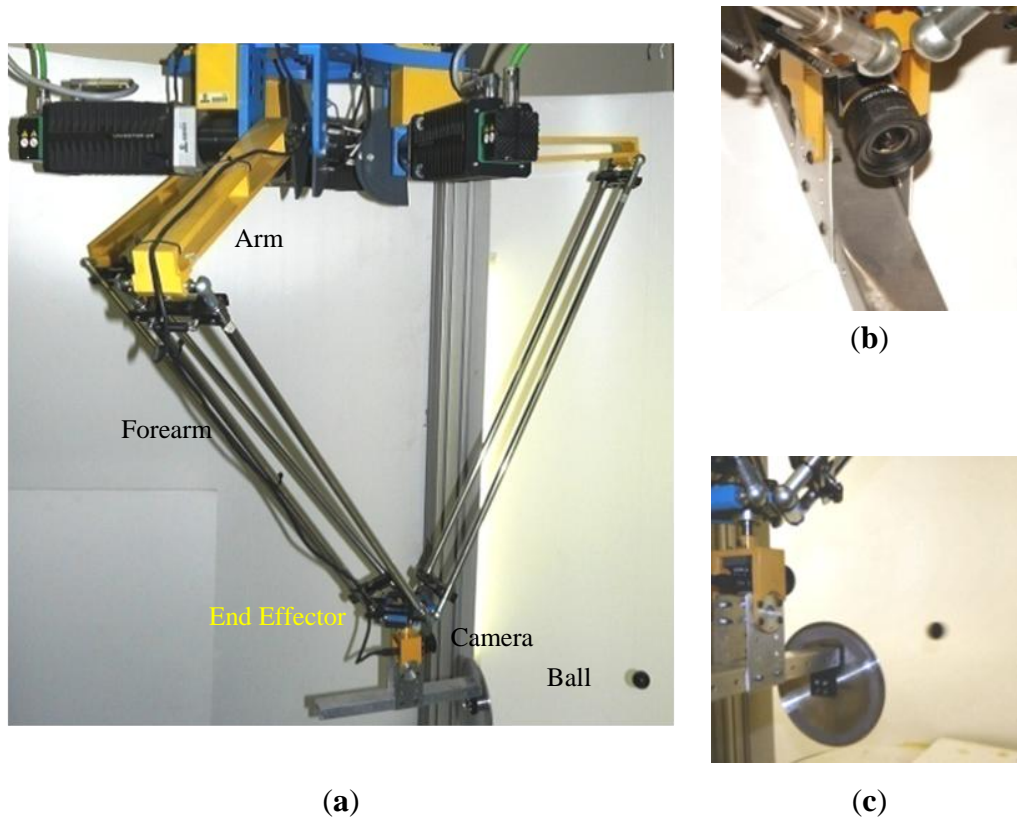
The main contribution of this approach is the use of low-cost sensor for calibrating the robot and the methodology used with the approval of the obtained results. Also it allows the evaluation of the position accuracy of homemade robot and the easy application to other robots developed in educational and research teams.

2. System Description

The main objective of the proposed method is to calibrate a parallel robot of the system called RoboTennis. The calibration of the robot is used for improving accuracy of its visual servoing algorithms and developing future tasks such as playing a game of ping-pong. The RoboTennis system was created to study and design visual servoing controllers and to carry out object tracking under dynamic environments. The mechanical structure of the RoboTennis system was inspired by the DELTA robot [34,35]. Its vision system is based on one camera allocated at the end effector of the robot. Basically, the RoboTennis platform (Figure 1a) consists of a parallel robot and a visual system for acquisition and analysis. The maximum end effector speed of the parallel robot is 6 m/s and its maximum acceleration is 58 m/s². The visual system of the platform RoboTennis [36] is made up of a 0.05 kg camera that is located at the end effector (Figure 1b,c) and a frame grabber (SONY XCHR50 and Matrox Meteor 2-MC/4 respectively). The motion system is composed of three AC brushless servomotors, AC drivers (Unidrive), and planetary gearboxes. The joint controller is implemented in a

DS1103 card (software implemented in ANSI C). The camera has been calibrated following the method proposed by Zhan [32].

Figure 1. (a) System RoboTennis, (b) Robot camera, (c) Robot environment.



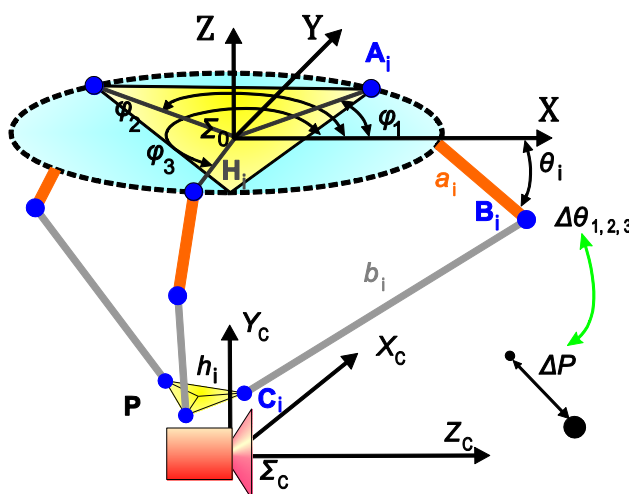
3. Constraint Equations

Constraint equations for the calibration method were obtained from a kinematic model of the robot. It is important to emphasize that the transformation matrix, between the camera coordinates system and the effector coordinates system, does not intervene in the equations of constraint used in the calibration of the robot, but it is obtained during the camera calibration process. Transformation matrix was divided into rotation and translation components. The rotation matrix was obtained independently from the robot calibration, whereas translation matrix is not necessary since the camera is fixed to the end effector and the model of the constraint equations is incremental. Figure 2 shows a sketch of the parallel robot, where:

- Σ_o is the robot reference frame
- i is the forearm and arm number, ($i \in \{1, 2, 3\}$)
- k is the incremental measurement number, ($k \in N$)
- A_i is the joint robot base arm, revolute axis of motor i
- B_i is the joint arm and forearm i , $B_i = [B_x \ B_y \ B_z]^T_i$
- C_i is the joint effector forearm i , $C_i = [C_x \ C_y \ C_z]^T_i$
- P is the centre of the end effector, $P = [P_x \ P_y \ P_z]^T$
- P_0 is the initial end effector position, $P_0 = [P_{0x} \ P_{0y} \ P_{0z}]^T$

- a_i is the arm length for the joint i
 b_i is the forearm i length
 h_i is the distance from P to C_i
 H_i is the distance from the robot reference frame Σ_o to A_i
 θ_i is the motor angle
 θ_{i0} is the initial motor angle (unknown)
 Φ_i is the angle in which point A_i is allocated in the XY plane of the robot reference frame

Figure 2. A sketch of the RoboTennis system.



The constraint equations are based on the movement of the forearms with respect to the robot arms [37]. In Delta robot, the forearms are parallelograms that link the end effector to the arms by means of spherical joints. Spherical joints allow modelling the movement of the end effector as the intersection of three spherical surfaces that are described by points C_i in Figure 3. By inspecting Figures 2 and 3, it is possible to obtain the surface of a sphere described by the point C_i with centre in B_i as:

$$\Gamma_i = (C_{ix} - B_{ix})^2 + (C_{iy} - B_{iy})^2 + (C_{iz} - B_{iz})^2 - b_i^2 = 0 \quad (1)$$

where B_i and C_i can be expressed in the robot frame, Σ_o as:

$$\begin{bmatrix} B_x \\ B_y \\ B_z \end{bmatrix} = \begin{bmatrix} (H + a \cos \theta) \cos \phi \\ (H + a \cos \theta) \sin \phi \\ a \sin \theta \end{bmatrix} \quad (2)$$

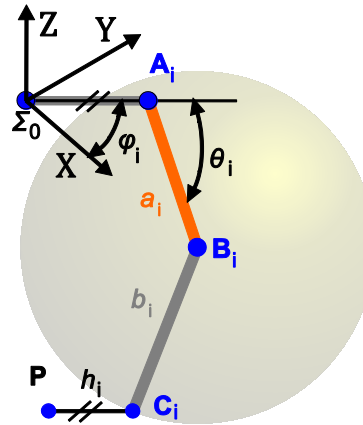
$$\begin{bmatrix} C_x \\ C_y \\ C_z \end{bmatrix} = \begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} - \begin{bmatrix} \cos \phi & -\sin \phi & 0 \\ \sin \phi & \cos \phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -h \\ 0 \\ 0 \end{bmatrix}$$

Since incremental joint and incremental end effector positions can be obtained (a similar approach for serial robots can be found in [38]), each absolute measurement can be expressed as:

$$\theta_{ik} = \theta_{i0} + \Theta_{ik} \quad \text{and} \quad P_k = P_0 + \Pi_k \quad (3)$$

Observe that θ_{i0} and P_0 are unknown but $\Theta_{ik} = \Delta\theta_{ik} = \theta_{ik} - \theta_{i0}$ and $\Pi_k = \Delta P_k = P_k - P_0$ are incremental positions and measured by the resolvers and the camera. So $V_k = [\Theta_1 \ \Theta_2 \ \Theta_3 \ \Pi_x \ \Pi_y \ \Pi_z]_k$ is a vector of measured parameters and it represents incremental positions of the end effector; they are measured from sphere images and k is the incremental measurement number.

Figure 3. One leg profile of the RoboTennis system.



4. Solution of the Constraints Equations

Not all of the robot parameters have an effect on the kinematic model and as a consequence they cannot be identified. This fact is known as observability of the term. An observability measurement can be derived from the Jacobian of the constraint equations. This Jacobian is called observation matrix and observable parameters can be obtained by its **QR** decomposition [39]. In the model, by means of the **QR** decomposition, the influences of the parameters Φ_i and H_i are difficult to observe. For this reason they are not identified in this work but the nominal value is taken into account. On the other hand, identifiable parameters are: a_i , b_i , h_i , and the reference positions of the actuators and the end effector of the robot: θ_{i0} and P_0 respectively (P_0 is the initial end effector position). Thus the unknown parameters can be expressed as: $u_i = [a_i \ b_i \ h_i \ \theta_{i0}]$ and $U = [u_1 \ u_2 \ u_3 \ P_{x0} \ P_{y0} \ P_{z0}]$

Substituting the incremental Equations (3) in Equations (2) and (1), the constraint equations can be arranged as:

$$\begin{aligned} \Gamma_{ik} = & (P_{x0} + h_i \cos\phi_i + \Pi_{xk} - (a_i \cos(\theta_{i0} + \Theta_{ik}) + H_i) \cos\phi_i)^2 \\ & + (P_{y0} + h_i \sin\phi_i + \Pi_{yk} - (a_i \cos(\theta_{i0} + \Theta_{ik}) + H_i) \sin\phi_i)^2 \\ & + (P_{z0} + \Pi_{zk} - a_i \sin(\theta_{i0} + \Theta_{ik}))^2 - b_i^2 = 0 \end{aligned} \quad (4)$$

In order to solve the above equations, Equation (4) is grouped as:

$$\Phi(U, V_k) = [\Gamma_{11} \dots \Gamma_{1k} \ \Gamma_{21} \dots \Gamma_{2k} \ \Gamma_{31} \dots \Gamma_{3k}]^T \quad (5)$$

Note that because the constraint equations are not linear, their solution is not exactly satisfied, thus, commonly, an approximate solution is obtained by numerical algorithms such as the Gauss-Newton method. Other possible alternatives are shown in [40]. By expressing the constraints in Equation (5) in them Taylor linear approximation (to be solved iteratively):

$$\Phi(U_{n+1}, V_k) \approx \Phi(U_n, V_k) + J(U_n, V_k)(U_{n+1} - U_n) = 0 \quad (6)$$

where n is the iteration number and $\mathbf{J}_{3k \times 15}$ is the Jacobian of the constraint equations or the observation matrix, which is given by:

$$\mathbf{J}(\mathbf{U}_n, \mathbf{V}_k) = \begin{bmatrix} \mathbf{J}_{1k} \\ \mathbf{J}_{2k} \\ \mathbf{J}_{3k} \end{bmatrix}$$

$$\mathbf{J}_{1k} = \begin{bmatrix} \frac{\partial \Gamma_{11}}{\partial a_1} & \dots & \frac{\partial \Gamma_{11}}{\partial \theta_{10}} & 0 & \dots & 0 & 0 & \dots & 0 & \frac{\partial \Gamma_{11}}{\partial P_{x0}} & \frac{\partial \Gamma_{11}}{\partial P_{y0}} & \frac{\partial \Gamma_{11}}{\partial P_{z0}} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{\partial \Gamma_{1k}}{\partial a_1} & \dots & \frac{\partial \Gamma_{1k}}{\partial \theta_{10}} & 0 & \dots & 0 & 0 & \dots & 0 & \frac{\partial \Gamma_{1k}}{\partial P_{x0}} & \frac{\partial \Gamma_{1k}}{\partial P_{y0}} & \frac{\partial \Gamma_{1k}}{\partial P_{z0}} \end{bmatrix}$$

$$\mathbf{J}_{2k} = \begin{bmatrix} 0 & \dots & 0 & \frac{\partial \Gamma_{21}}{\partial a_2} & \dots & \frac{\partial \Gamma_{21}}{\partial \theta_{20}} & 0 & \dots & 0 & \frac{\partial \Gamma_{21}}{\partial P_{x0}} & \frac{\partial \Gamma_{21}}{\partial P_{y0}} & \frac{\partial \Gamma_{21}}{\partial P_{z0}} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \dots & 0 & \frac{\partial \Gamma_{2k}}{\partial a_2} & \dots & \frac{\partial \Gamma_{2k}}{\partial \theta_{20}} & 0 & \dots & 0 & \frac{\partial \Gamma_{2k}}{\partial P_{x0}} & \frac{\partial \Gamma_{2k}}{\partial P_{y0}} & \frac{\partial \Gamma_{2k}}{\partial P_{z0}} \end{bmatrix} \quad (7)$$

$$\mathbf{J}_{3k} = \begin{bmatrix} 0 & \dots & 0 & 0 & \dots & 0 & \frac{\partial \Gamma_{31}}{\partial a_3} & \dots & \frac{\partial \Gamma_{31}}{\partial \theta_{30}} & \frac{\partial \Gamma_{31}}{\partial P_{x0}} & \frac{\partial \Gamma_{31}}{\partial P_{y0}} & \frac{\partial \Gamma_{31}}{\partial P_{z0}} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \dots & 0 & 0 & \dots & 0 & \frac{\partial \Gamma_{3k}}{\partial a_3} & \dots & \frac{\partial \Gamma_{3k}}{\partial \theta_{30}} & \frac{\partial \Gamma_{3k}}{\partial P_{x0}} & \frac{\partial \Gamma_{3k}}{\partial P_{y0}} & \frac{\partial \Gamma_{3k}}{\partial P_{z0}} \end{bmatrix}$$

Finally the parameters are iteratively obtained by:

$$\mathbf{U}_{n+1} = \mathbf{U}_n - \mathbf{J}(\mathbf{U}_n, \mathbf{V}_k)^+ \Phi(\mathbf{U}_n, \mathbf{V}_k) = 0 \quad (8)$$

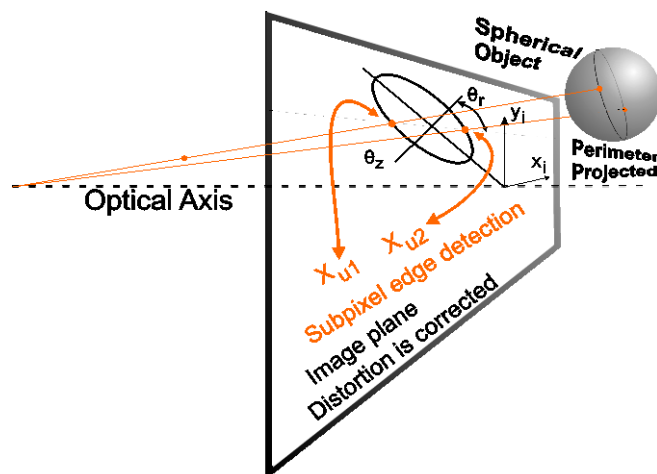
where, \mathbf{J}^+ is the pseudo inverse Jacobian. Initial values (\mathbf{U}_0) are given by the nominal parameters and a pose of the robot (given from nominal parameters). In the constraint equations in Equation (4), the units of the Jacobian (7) are homogeneous. There is not need to normalize the Jacobian.

5. Visual Measurement System

The visual measurement system is made up of a fixed (by a transparent rod) ball and a calibrated camera that is located at the end effector of the robot. The visual system has to get the relative robot position with respect to the ball in order to deliver values used in Equation (8) to the joint incremental and the end effector position. In this section, various fundamental points are analyzed to reach a good accuracy in the parameters from the robot calibration. On the one hand, it is necessary to set the wanted end effector positions to their number. On the other hand, it is also necessary to get the visual data with the greatest accuracy as possible. Besides using subpixel information, it has been considered important to correct two ball projection distortions by mean of a novel algorithm. The projected sphere is seen as an ellipse in the image. A first effect to correct is that the ellipse does not match with the projection of a ball section with the real ball centre. The second effect is the difference between the

axis ellipse extremes and the values obtained in the horizontal and vertical lines that cross the ellipse centre (see Figure 4).

Figure 4. Projection of the ball in the image plane.



5.1. Selection of Calibration Poses and Obtained Robot Parameters

In order to solve the system of constraint equations, a set of incremental positions (resolvers and end effector) is needed. Several simulations have shown that at least 6 different perfect pose measurements are required. However, it is obvious that perfect measurements cannot be obtained. Therefore, it is necessary to choose best calibration poses. Thus, errors in the parameters of the robot are especially critical when the calibration pose is near a singularity. However in a practical calibration process, if poses are extremely close to singularities, they can produce unpredictable movements (due to kinematic errors). In this work, the robot workspace is numerically well known, by using nominal values, and the calibration poses are chosen randomly from the workspace boundaries. Thus if a pose is located between 5 and 10 cm from a singularity, then it is included as a calibration pose. The experiments were implemented with 300 poses and each measure is the mean of 500 images in order to avoid noise from image acquisition (image acquisition rate: 8.34 ms) 21 minutes are needed for the acquisition of all images. Additionally, it is necessary to wait 4 seconds at each change of pose in order to ensure the absence of any vibration in the robot or the holding structure. The image processing is performed while the acquisition process, while the time taken by the optimization algorithm is almost negligible compared to the rest (about 5 seconds). Due to its simplicity, the calibration process is done automatically in 40 minutes and it can be calibrated on-line. Several calibrations have been executed and then several sets of calibrated parameters have been used in the initial tests. Since similar results have been obtained, the set of calibrated parameters that produces less error in the tests (see Section 6) has been chosen to compare with the nominal parameter in the rest of the tests. In Table 1, both parameter sets are shown.

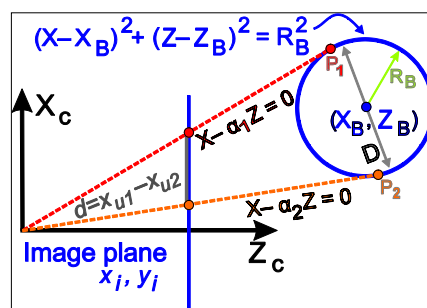
Table 1. Robot kinematic parameters, nominal and calibrated.

	Nominal			Calibrated		
	i = 1	i = 2	i = 3	i = 1	i = 2	i = 3
a_i (mm)	500	500	500	500.89	500.01	499.96
b_i (mm)	1,000	1,000	1,000	1,002.86	1,001.37	1,001.28
h_i (mm)	50	50	50	48.69	49.96	48.68
θ_i (degrees)	0	0	0	0.316	0.844	0.936
P_x (mm)		0			5.31	
P_y (mm)		0			0.31	
P_z (mm)		750			743.74	
ϕ_i (degrees)	0	120	240	0	120	240
H_i (mm)	210	210	210	210	210	210

Parameters that were not estimated.

5.2. Three-Dimensional Position Correction

As previously mentioned, the aim is to obtain the position of the centre of the ball, (X_B, Y_B, Z_B) shown in the camera coordinates, from the acquired visual information. With the data from the kinematic camera calibration, the relative position between the robot and the fixed ball is determined. In this section, an algorithm that makes a first correction of the ball projection in order to increase the accuracy of the detection is detailed out. The projected sphere is seen as an ellipse in the image. Nevertheless, the ellipse does not match with the projection of a ball section with the real ball centre. This little difference thereby affects the obtained position. In the method description a 2D model as shown in Figure 5 is used. The plane X_c - Z_c (camera reference system) that contains the sphere centre ($Y_c = Y_B$) is shown. On the other hand, each point in the image is related by a line that passes through the optical centre of the camera. In this case, the interesting points are the points that belong to the two diametrically opposed tangents to the sphere.

Figure 5. Simplified projection of the ball in the image plane.

The image points, x_{u1} and x_{u2} , in central images coordinates correspond with the projection of the points P_1 and P_2 , whose distance is not the ball diameters strictly. So:

$$X_1 = \alpha_1 Z_1 \quad \text{and} \quad X_2 = \alpha_2 Z_2 \quad (9)$$

The values α_1 and α_2 are obtained from the projection of the tangential points to the sphere, x_{u1} and x_{u2} . It will be:

$$\frac{x_{u1}}{f} = \frac{X_1}{Z_1} = \alpha_1; \quad \frac{x_{u2}}{f} = \frac{X_2}{Z_2} = \alpha_2 \quad (10)$$

where, f is the focal distance (determined in the camera calibration process). The sphere is fixed in the space, its radio is known ($R_B = 19$ mm), and the distance of the line that is tangent to the perimeter to the centre of the sphere (X_B, Y_B, Z_B) is given by:

$$R_B = \frac{|X_B - \alpha_{1,2}Z_B|}{\sqrt{1 + \alpha_{1,2}^2}} \quad (11)$$

The term $(X_B - \alpha_{1,2}Z_B)$ has different sign in the extreme values (it is null for values near to the mean value), then following equations are possible:

$$\begin{aligned} -X_B + \alpha_1 Z_B &= R_B \sqrt{1 + \alpha_1^2} \\ X_B - \alpha_2 Z_B &= R_B \sqrt{1 + \alpha_2^2} \end{aligned} \quad (12)$$

From here it can obtain X_B and Z_B directly but it is necessary to take the Y_c - Z_c plane information. So, the same way in this plane the tangential points will fulfil:

$$Y_1 = \beta_1 Z_1 \quad \text{and} \quad Y_2 = \beta_2 Z_2 \quad (13)$$

The values of β_1 and β_2 are got from the projection of the tangential points to the sphere, y_{u1} and y_{u2} . It will be:

$$\frac{y_{u1}}{f} = \frac{Y_1}{Z_1} = \beta_1 \quad ; \quad \frac{y_{u2}}{f} = \frac{Y_2}{Z_2} = \beta_2 \quad (14)$$

Likewise, considering the distance from the sphere centre to each tangent line:

$$R_B = \frac{|Y_B - \beta_{1,2}Z_B|}{\sqrt{1 + \beta_{1,2}^2}} \quad (15)$$

With similar reasons that for Equation (12):

$$\begin{aligned} -Y_B + \beta_1 Z_B &= R_B \sqrt{1 + \beta_1^2} \\ Y_B - \beta_2 Z_B &= R_B \sqrt{1 + \beta_2^2} \end{aligned} \quad (16)$$

Then, with Equations (12) and (16) there are four equations with three unknowns, (X_B, Y_B, Z_B), that can be solved using the least squares method.

5.3. Ellipse Ball Projection Shape Correction

The second developed correction allows calculating the ellipse axes (this ellipse is the projection of the sphere on the image) from the obtained values in the horizontal and vertical lines with intersection in the ellipse centre (see Figure 6). In Figure 6, take note that x_{u1} and x_{u2} are not on the axis (the same for the points y_{u1} and y_{u2} their use thereby influences negatively in the determination of the sphere position (according to it is described in Section 5.2). The objective is to obtain the axes major and minor of the projected ellipse in the image (a_e, b_e) from the known information (x_{u1} and x_{u2}). Once the

semi-minor axis of the projected ellipse is calculated, the algorithm seen in Section 5.2, in order to calculate the centre of the sphere definitively, is used.

Figure 6. Ellipse of the sphere projection on the image plane.

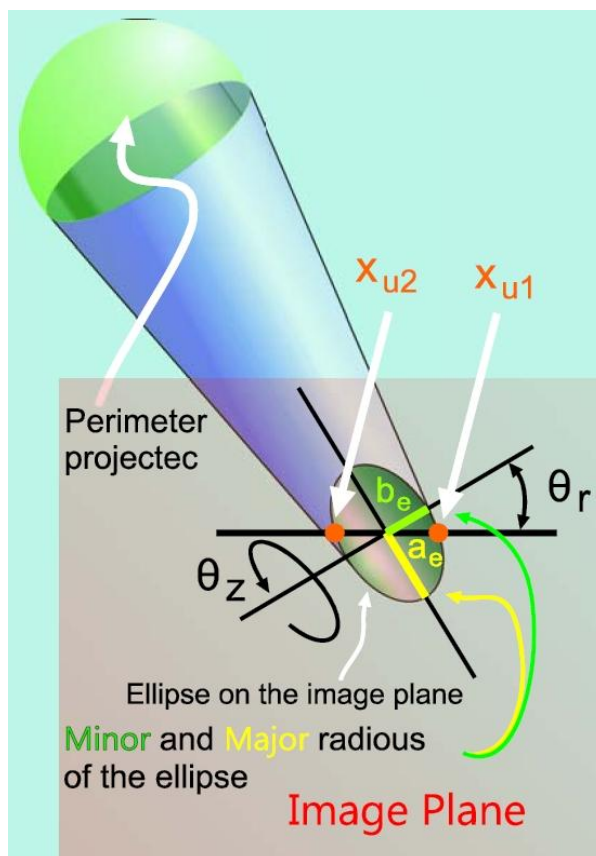


Figure 6 shows the error due to projection deformation; it can be modelled by knowing the angle θ_z and the object position along the axis X_C . θ_z is the angle between the optical axis and the line that passes through the centre of the ball and the centre of the projected ball. Note that θ_z coincides with the angle by where the image plane can be rotated around the minor axis of the ellipse in order to project a circle on the plane image, thus:

$$\theta_z = a \cos \left(\frac{Z_B}{\sqrt{(X_B^2 + Y_B^2 + Z_B^2)}} \right) \quad (17)$$

where (X_B, Y_B, Z_B) is the previously estimated position of the sphere centre in the camera coordinates system. On the other hand, the minor (b_e) and major (a_e) axes of the projected ellipse are related by:

$$b_e = a_e \cos(\theta_z) \quad (18)$$

Observing Figure 6, it is deduced that the rotated ellipse is modelled suitably with the canonical parametric model. The projected ellipse in its rotated canonical parametric form can be expressed as:

$$\begin{aligned} x_{ue} &= b_e \cos(\theta_e) \cos(\theta_r) - a_e \sin(\theta_e) \sin(\theta_r) + x_{uc} \\ y_{ue} &= b_e \cos(\theta_e) \sin(\theta_r) + a_e \sin(\theta_e) \cos(\theta_r) + y_{uc} \end{aligned} \quad (19)$$

where θ_e is the angle of a point in the canonical form of the ellipse, θ_r is the angle by which the ellipse is rotated around the Z axis of the camera (Equation (19)) and $(x_{uc} \ y_{uc})$ is the centre of the ellipse in the image, Figure 7.

$$\theta_r = \arctan\left(-\frac{y_{uc}}{x_{uc}}\right) \quad (20)$$

The ellipse point in the ellipse horizontal line that pass by the centre satisfies (see Figure 7) $x_{uc} = x_{ue} + (x_{u2} - x_{u1})/2$; $y_{ue} = y_{uc}$. By substituting in Equation (19) it is possible to obtain:

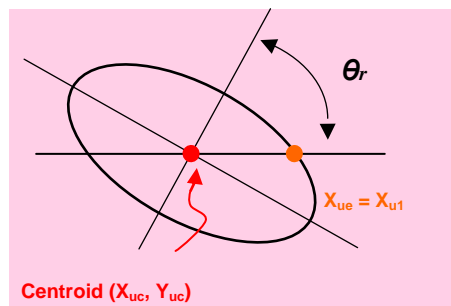
$$\begin{aligned} (x_{u2} - x_{u1})/2 &= b_e \cos(\theta_e) \cos(\theta_r) - a_e \sin(\theta_e) \sin(\theta_r) \\ 0 &= b_e \cos(\theta_e) \sin(\theta_r) + a_e \sin(\theta_e) \cos(\theta_r) \end{aligned} \quad (21)$$

By substituting Equation (18) in (21) and clearing for a_e and θ_e finally it is obtained:

$$\begin{aligned} \theta_e &= \arctan(-\tan(\theta_r) \cos(\theta_z)) \\ 0 \ a_e &= \left| \frac{(x_{u2} - x_{u1})/2}{\cos(\theta_z) \cos(\theta_e) \cos(\theta_r) - \sin(\theta_e) \sin(\theta_r)} \right| \end{aligned} \quad (22)$$

With the value of a_e , b_e is calculated and setting in Equation (19) the values of $\theta_e = 0, \pi/2, \pi, \pi/2$, allows us to determinate the new values for the extreme points used in the Equations (10) and (14); afterwards, the sphere centre position is calculated with the Equations (12) and (16). And so on. It could continue calculating with the new data this algorithm, but it has been checked that the improvement is not appreciable after a second iteration.

Figure 7. Angle θ_r .



5.4. Summary of the Proposed Calibration Method

The proposed calibration algorithm is executed according to the following steps:

- I. Initialize the actuators of the robot in a home position (θ_{i0} , that it is taken as unknown in the calibration algorithm).
- II. Randomly, inside of the workspace (not necessary well conditioned but near of the workspace boundaries) of the robot, generate a set of 300 poses of the robot to be used as calibration poses.
- III. The home of the robot is taken as unknown (θ_{i0} and \mathbf{P}_0), but incremental measurements are acquired from: encoders of the actuators (incremental angles, θ_{ik}) and from the visual system (incremental poses, \mathbf{II}_k).

- a. Acquire 500 images in each pose of the robot (using the set in II) and by using image processing from each image α_1 , α_2 , β_1 , and β_2 are obtained. By using Equations (12) and (16) obtain (X_B, Y_B, Z_B) is obtained.
 - b. In order to correct the projection of the spherical object in the image, use Equation (22) to correct (X_B, Y_B, Z_B) and repeat above step once.
- IV. Obtain new parameters of the robot by using above incremental measurements and equation in (8). It is well known that Newton-Raphson method could not converge to useful value. Thus result should be tested.

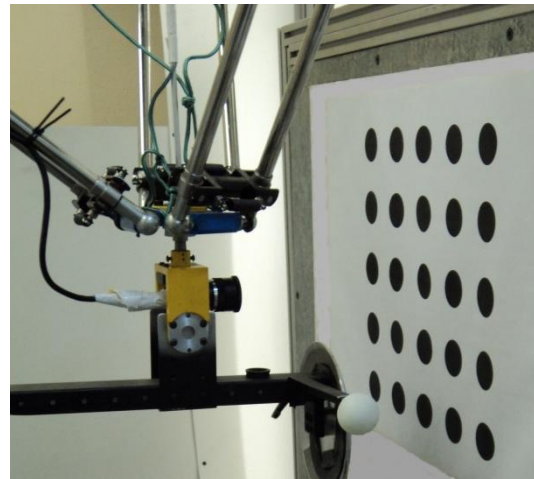
6. Results

A set of test cases is implemented to check the goodness of the parameters obtained by the calibration algorithm. The behaviour between a calibrated and a not calibrated robot system is compared. Two types of tests, static and tracking tests have been implemented. By executing static tests, it is determined how calibration parameters affect the robot positioning. By the second type, it is determined how calibration parameters affect the tracking tasks in which the RoboTennis has been designed.

6.1. Static Tests

The objective of the static tests is to quantify the effect over the robot positioning by using the parameters obtained from the robot calibration process against the nominal parameters. Both parameter sets are shown in Table 1. Furthermore, the tests contribute significant information about the system behaviour. The test environment is made up of the RoboTennis (the system to test), a vision camera (measurement system) and a control element. Following the same idea in the robot calibration algorithm, the vision camera fixed to the robot hand in order to see the fixed objects in the robot visual area, is used as lonely measurement too. In order to simplify, the control element is two-dimensional formed by a plane surface with 25 black circles of 40 mm of diameter separated by 100 mm to each other, see Figure 8. A three-dimensional control element would have complicated the process without a significant information contribution. The circles have been printed on high quality and accuracy and the paper is glued to a rigid plate to make a plane exact. This control element is placed on vertical places in different distances from the robot and on a parallel plane to the image plane in each test, as see in Figure 8.

In order to evaluate the results in the robot motion, the data from the visual system when it sees the control element have been analyzed. Indeed, the repeatability of the three-dimensional data, when a circle to 200 mm of distance is seen by the camera in the image centre (to minimize the camera calibration errors) with the robot blocked, has also been analyzed. The visual resolution is 0.4 mm per pixel approximately in the horizontal axis. The circle position is determined by analysis of the circle rim with subpixel precision. Other results are the centre and diameter. A loop of 50 samples with the robot blocked allows the repeatability of the measurement to be obtained. Some representative data are shown in Table 2. It is important to emphasize the difference between X_c and Y_c coordinates (image plane), due to a different camera resolution and a bigger uncertainty in the camera Z_c axis since it depends on the circle diameter.

Figure 8. RoboTennis system and control element.**Table 2.** Standard deviation and Range (max value–min value) of 50 samples.

Axis	Standard deviation	Range (mm)
X_c	0.014	0.059
Y_c	0.038	0.163
Z_c	0.092	0.497

The high variation of the data shown in Table 2 means; it is not possible to use any visual measurement to estimate the robot position with high accuracy. In the static measurements, the mean of 500 samples of the circle is used because it is significantly more accurate. Standard deviation and Range for 25 measurements calculated one by one from 500 samples are shown in Table 3.

Table 3. Standard deviation and Range for 25 measures analyzed.

Axis	Standard Deviation	Range (mm)
X_c	0.0025	0.0096
Y_c	0.0049	0.0150
Z_c	0.0212	0.0846

The repeatability of the measurement according to the ISO 9283 [41] (see Appendix) is 0.0635 mm. The data obtained with other test sets are similar.

The next element to be considered is the robot system itself. The maximum accuracy depends on the considered working point. With the used encoders, it is not possible to guarantee an accuracy that is less than 0.02 mm. In order to determine the robot position, the camera is placed 200 mm of distance. in front of each circle The robot analyzes the image and fixes its position till the circle centre matches the image centre, with an error less than 1 pixel (to minimize the camera calibration errors). The robot position data from the encoders are corrected with the vision system data. As previously described, in order to minimize the error in the visual data acquisition, a reliable measurement for the mean of 500 snapshots and simple data processes are considered. In such manner, accurate calculations of robot position with respect to the control element can be obtained.

6.1.1. Positioning Repeatability Analysis

Only positioning repeatability has been analyzed as the first analysis. Position accuracy depends on the calibration parameters used. It is the main aim of this research and is analyzed in Section 6.1.2 with incremental data. Orientation repeatability and accuracy have not been analyzed because the robot system has only three degrees of freedom, all of them being translation.

According to the ISO 9283 protocol for positioning repeatability, a plane in a diagonal of a cube into the robot working area is chosen. In the plane, five points are chosen (the vertices of a rectangle and its centre). The robot has to execute a loop formed by positioning in front of each point and repeatable measure the last one (focus point). For such, as mentioned above, the control element is placed 200 mm with a black circle in the image centre. The visual system analyzes the three-dimensional position of the circle. After 500 samples, the position of the robot is known.

In Table 4, the obtained values are shown after the positioning loop in the five points, have been executed 25 times. The results are similar and do not depend on the focused point (central point results in Table 4). The tests have been done with two robot speeds, 1.0 m/s and 2.0 m/s, with a negligible load (camera and bat). Table 4 shows the range of the visual data in 25 samples and the repeatability calculated according to the ISO 9283 (see annex A). The data are coherent with the robot accuracy (0.02 mm) and uncertainty visual data (0.0635).

Table 4. Repeatability of position, loop 25 poses.

Speed (m/s)	Visual Data Range (X_c , Y_c , Z_c) with 25 Samples (mm)	Repeatability of Position (mm)
1.0	0.032, 0.025, 0.114	0.07041
2.0	0.039, 0.034, 0.121	0.07512

As conclusion, it can be affirmed that the robot system and the visual sensor have repeatability of less than 0.1 mm under normal working conditions (space working points, robot speed, and load). This value has to be quite small enough so as not to affect in the accuracy of the positioning distance (next section).

6.1.2. Accuracy of the Positioning Distance Analysis

The aim of the following tests is to quantify the produced effect in the accuracy robot positioning by the parameters obtained in the calibration robot process against the nominal parameters. Both parameter sets are shown in Table 1. In order to obtain such, again, the vision system and control element have been used once more. Since no external sensor is used neither an external absolute reference system is needed, so it is more representative to analyze the accuracy of the positioning distance. The control system has been placed in five fixed vertical planes which take a homogeneous working robot area (see Figures 9 and 10). Each plane is placed parallel to the camera plane, perfect parallelism is not required. The distance between the planes is 100 mm. It does not need to be perfectly positioned since the data to be analyzes are the distances between the circles of the same plane. The analyzed area corresponds to a cube of 400 mm in each side. This physical configuration allows studying the accuracy of the positioning distance as function of the position inside the working area.

Figure 9. Spatial distribution of the obtained errors by using calibrated parameters. The mean of all errors is 0.583 mm (dimension values in mm).

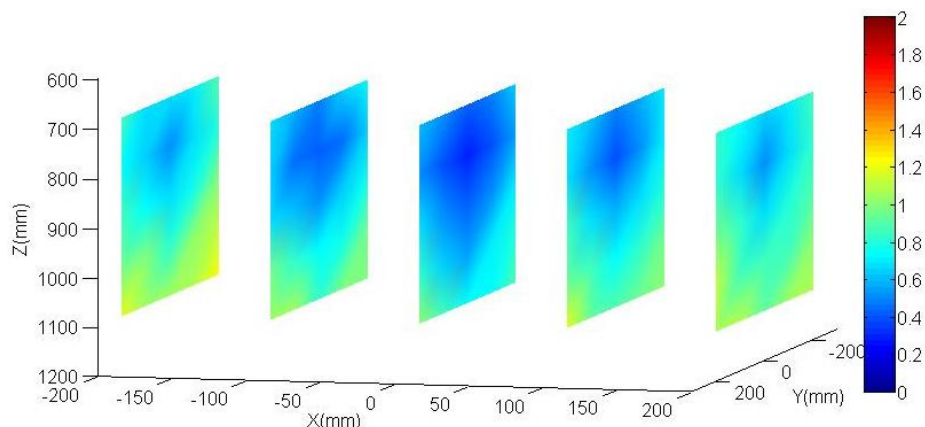
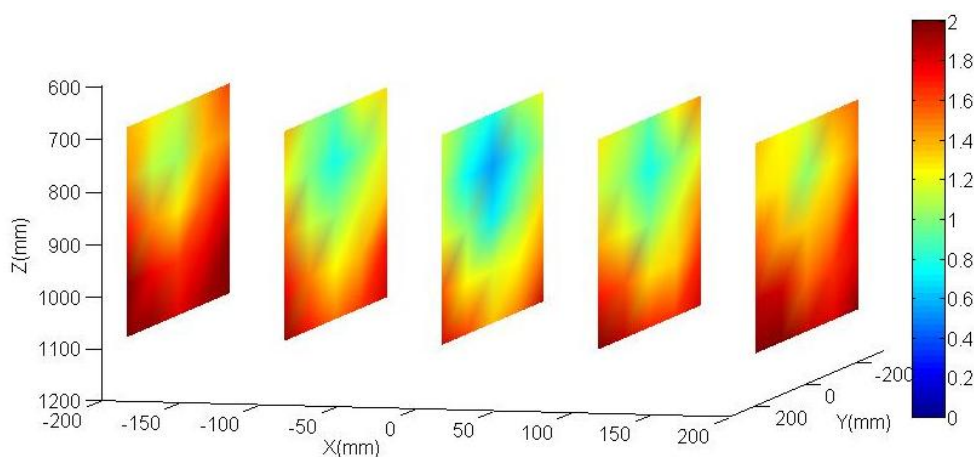


Figure 10. Spatial distribution of the obtained errors by using nominal parameters. The mean of all errors is 1.468 mm (dimension values in mm).



The robot positioning in relation to the control element is done likewise as previously mentioned. The robot is placed 200 mm in front of each circle, the position is fixed till the circle is in the image centre, and the three-dimensional data position corrected with the visual information is stored. Its accuracy is considered well known to measure the mean of 500 samples.

For each circle to be analyzed (five planes with 25 circles each one), the distance between the circle and its neighbour in the same plane has been calculated by getting the absolute error. The absolute error is calculated by comparing the obtained distance with the theoretical distance of the control element. For this study a loop of 25 samples has been done according to the ISO 9283 (see Appendix). As there are several distances (so many as neighbours). The mean of the error in every distance has been calculated.

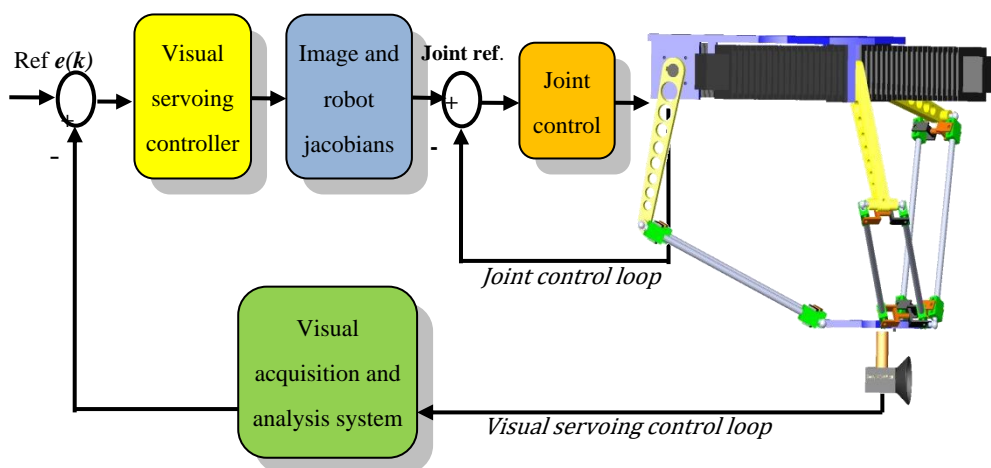
Figures 9 and 10 show results obtained both for the calibrated parameters and the nominal parameters. The error is rendered by false colour, scale between 0 and 2 mm. For each one out of five planes distributed along the working area (in the XYZ coordinate system, located at the base of the robot, Figure 2), an interpolation has been done between the obtained results for each circle of the plane. Also the mean of all the errors is shown. As it can be observed, the errors obtained in the

positioning distance decreased almost 40% using calibrated parameters. This justifies the efforts invested in the robot calibration. Among other reasons why it is not possible to minimize that error any farther, it is important to highlight that the method does not calibrate all robot parameters. It can be also observed that the errors increase in the working area limits.

6.2. Tracking Test

The objective of these tests is to quantify how calibration parameters affect the tracking tasks. It is desired that the robot is able to track an object that moves chaotically in its workspace. In order to test the calibration method, nominal parameters and calibrated parameters are compared when they are utilized in a position-based visual controller [42]. The robot and its visual controller are designed to carry out tracking tasks. For more information, please consult [43]. The visual controller is based on a well established architecture called: dynamic position-based look-and-move visual servoing. In this scheme (Figure 11), the visual controller makes use of image estimates (position and velocity), the Jacobian of the robot and its kinematic model in order to act over the robot actuators. Thus errors in the kinematic model generate undesirable movements that have an effect on the performance of the visual controller.

Figure 11. Visual servoing controller scheme of the RoboTennis System.



First this has been studied by simulating the effect it can have on tracking tasks using erroneous values of the parameters of the robot. It seeks to limit its maximum theoretical influence. Later follow-up tests are real, using both nominal and calibrated values.

6.2.1. Simulation of the Influence of the Use of Incorrect Parameters

The visual control law based on position can be defined [42] as:

$$\hat{\mathbf{u}}_r = -K_p \mathbf{e}_r + \hat{\mathbf{v}}_p \quad (23)$$

where:

K_p is a positive scalar which fits experimentally (for test 16.8);

\mathbf{e}_r is the error in Cartesian coordinates, defined as the difference between the actual relative position of the object to the robot, and the desired relative position. It is obtained from visual features;

$\hat{\mathbf{v}}_p$ is the estimated speed of the monitored object.

If the direct Jacobian robot [26] is defined as:

$$\dot{\mathbf{r}} = \mathbf{J}_R \dot{\mathbf{q}} \quad (24)$$

where:

Articulate position: $\mathbf{q} = [q_1 \quad q_2 \quad q_3]^T$

Cartesian position: $\mathbf{r} = [X \quad Y \quad Z]^T$ (absolute position of the robot, Figure 2).

The RoboTennis system, like the Delta robot, has a Jacobian as a 3×3 matrix, which supports reverse except singular points.

The joint control law, which is acting on the engines, is:

$$\hat{\mathbf{u}}_q = \hat{\mathbf{J}}_R^{-1} \hat{\mathbf{u}}_r = \hat{\mathbf{J}}_R^{-1} (-K_p \mathbf{e}_r + \hat{\mathbf{v}}_p) \quad (25)$$

where $\hat{\mathbf{J}}_R$ is the estimated Jacobian, which can be obtained from or nominal or calibrated values. This Jacobian is the one used in the implementation.

The Cartesian robot behaviour (without considering the dynamics of the system) depends on the real robot Jacobian, which is unknown. It corresponds to what would move the robot when the motors are actuated.

$$\mathbf{u}_r = \mathbf{J}_R \hat{\mathbf{J}}_R^{-1} \hat{\mathbf{u}}_r = \mathbf{J}_R \hat{\mathbf{J}}_R^{-1} (-K_p \mathbf{e}_r + \hat{\mathbf{v}}_p) \quad (26)$$

In order to analyze the influence of the use of nominal or calibrated parameters in the control law, it is possible to compare the difference in the movement of the robot between when using a control law with nominal parameters or when using calibrated. If defined:

$\hat{\mathbf{J}}_{RC}$ Direct robot Jacobian obtained from calibrated parameters

$\hat{\mathbf{J}}_{RN}$ Direct robot Jacobian obtained from nominal parameters

$\mathbf{J}_R = \hat{\mathbf{J}}_{RC}$ Considered the best estimation

It is obtained:

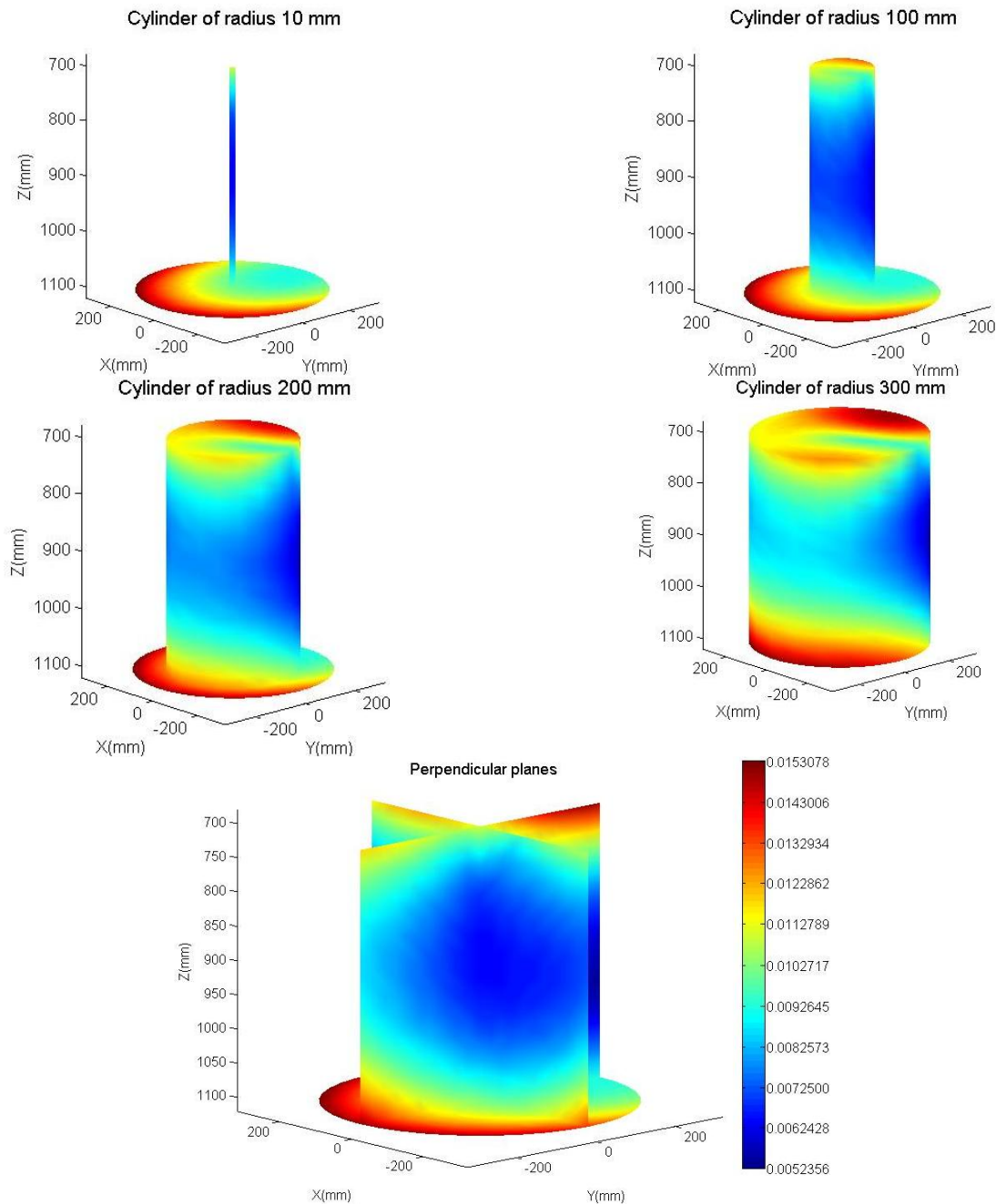
$$\Delta \mathbf{u}_r = \mathbf{u}_{rN} - \mathbf{u}_{rC} = \hat{\mathbf{J}}_{RC} (\hat{\mathbf{J}}_{RN}^{-1} - \hat{\mathbf{J}}_{RC}^{-1}) (-K_p \mathbf{e}_r + \hat{\mathbf{v}}_p) \quad (27)$$

Also:

$$\Delta \mathbf{u}_r = \hat{\mathbf{J}}_{RC} (\hat{\mathbf{J}}_{RN}^{-1} - \hat{\mathbf{J}}_{RC}^{-1}) \mathbf{v} = (\hat{\mathbf{J}}_{RC} \hat{\mathbf{J}}_{RN}^{-1} - \mathbf{I}) \mathbf{v} \quad (28)$$

where \mathbf{v} is a generic input speed. It is necessary to analyze the influence of the term $\Delta \hat{\mathbf{J}}_R = \hat{\mathbf{J}}_{RC} (\hat{\mathbf{J}}_{RN}^{-1} - \hat{\mathbf{J}}_{RC}^{-1})$ at different velocities. Figure 12 shows a spatial distribution of the largest singular value of said matrix into a cylinder of radius 300 mm and height 400 mm, included in the workspace. The absolute coordinates of the robot, (X, Y, Z), are used. The largest singular value is the spectral norm of the matrix and it represents, in that matrix, the upper limit of how the change speed influences in the control law.

Figure 12. Largest singular value of the matrix $\Delta \hat{\mathbf{J}}_R$ in a cylinder of radius 300 mm and height 400 mm, included in the workspace.



For a better visualization, the largest singular value along the exterior of various cylinders (10 mm, 100 mm, 200 mm, and 300 mm in radius), and in two perpendicular planes is represented in this figure with false colour. The last figure (on the right) shows the colorimetric scale used. The observed asymmetry is due to the lack of symmetry of the calibrated parameters and its difference from the nominal values. The simulation environment has avoided the singular points of both Jacobians, those points alter the study. The actual control avoids such singular points obtained using the Jacobian at a near no singular point.

Assuming maximum values:

$$\|\mathbf{e}_r\| = 30 \text{ mm}$$

$$\|\hat{\mathbf{v}}_p\| = 1500 \text{ mm/s}$$

$$\|\mathbf{v}\| = 16.8 * 30 + 1500 \approx 2000 \text{ mm/s}$$

$$\|\hat{\mathbf{J}}_{RC}(\hat{\mathbf{J}}_{RN}^{-1} - \hat{\mathbf{J}}_{RC}^{-1})\| = 0.015$$

It is obtained:

$$\|\Delta \mathbf{u}_r\| \approx 0.015 * 2000 \approx 30 \text{ mm/s}$$

It represents approximately 1.5% difference from the control law, which is the greatest influence that would be obtained by using the parameters calibrated against nominal values.

6.2.2. Tracking Real Tests

In order to analyze the influence of a best calibration on the robot parameters, several tracking tests of a ball with a chaotic movement at speeds up to 1,500 mm/s have been done. The objective is to keep the robot at a constant distance of 600 mm with the ball in the image centre. In Table 5, the mean of the tracking errors in millimetres when 10 tests are done for the calibrated parameters and for the nominal parameters is shown, in function of the speed of the ball. Each test consists of 5,000 samples. A certain improvement, in a smaller proportion compared to the one indicated in Section 6.1.2, can be observed. This aspect can be explained by the high tolerance in the image characteristics-based control with the influence of the calibration errors.

Table 5. Tracking Error Mean (mm).

Maximum Speed of the Ball	Error with Nominal Parameters	Error with Calibrated Parameters
500 mm/s	4.54	4.47
750 mm/s	7.42	7.32
1,000 mm/s	12.76	12.61
1,250 mm/s	20.14	19.84
1,500 mm/s	30.47	30.14

7. Conclusions

In this paper, a calibration method using a camera attached to the robot hand as the only main sensor in order to improve the accuracy of a parallel robot inspired by the Delta robot is described. The simplicity of the equipment used over other more expensive alternatives, makes this proposal especially useful for calibration of robots in educational and research environments. The methodology for visual data acquisition is especially accurate, and can be extended to other applications.

The improvements using this method have been verified by executing several sets of tests (static and tracking tests). Before these tests, the measure repeatability of the whole system, which is close to 0.0635 mm according to [41], has been analyzed.

From the first static tests, it can be affirmed that the positioning repeatability of the robot system and the visual sensor is less than 0.1 mm under normal working conditions (space working points, robot speed, and load). From the second kind of static tests, it can be pointed out that using calibrated parameters reduces the errors in the positioning distance by almost 40%, which is a remarkable error reduction.

On the other hand, the tracking tests, by defining an index depending on the errors in the controller and an estimated working velocity, both sets (nominal and corrected) of kinematic parameters have been compared through the visual servoing controller performance. The increase in the performance obtained from the tracking visual controller regarding the correction of the parameters is extremely minimal (errors in the controller are reduced by more than 1.5% when the robot is calibrated). The improvements in the controller performance can be explained as follows: The image visual controller makes use of the Jacobian robot and kinematic robot models (inverse and direct). When the controller corrects the robot trajectory, errors in the model have a direct influence over the direction in which the controller tries to correct. On the other hand if the position of the object remains fixed, the error in the visual controller converges to zero but the evolution of the error is different for two set of parameters.

One aspect to analyze in future work is the incorporation of non-calibrated parameters in the optimization algorithm in order to further minimize the errors obtained. Some videos of the RoboTenis system in tracking tasks (such as playing ping-pong) are shown at: <http://www.disam.upm.es/vision/projects/robotenis/indexI.html>.

Acknowledgements

The authors would like to thank the following institutions for their financial support: Universidad Politécnica de Madrid in Madrid Spain; Universidad Aeronáutica de Querétaro, in Queretaro Mexico; and Universidad Nacional de San Juan in San Juan, Argentina. This work was supported by the Ministerio de Ciencia e Innovación of the Spanish Government under Project DPI2010-20863

Conflict of Interest

The authors declare no conflict of interest.

References

1. Ma, D.; Hollerbach, J.M.; Xu, Y. Gravity Based Autonomous Calibration for Robot Manipulators. In Proceedings of the 1994 IEEE International Conference on Robotics and Automation, San Diego, CA, USA, 8–13 May 1994; pp. 2763–2768.
2. Ecorchard, G.; Maurine, P. Self-Calibration of Delta Parallel Robots with Elastic Deformation Compensation. In Proceedings of the 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems, Edmonton, AB, Canada, 2–6 August 2005; pp. 1283–1288.
3. Zhang, D.; Gao, Z. Optimal kinematic calibration of parallel manipulators with pseudoerror theory and cooperative coevolutionary network. *IEEE Trans. Ind. Electron.* **2012**, *59*, 3221–3231.
4. Majarena, A.C.; Santolaria, J.; Samper, D.; Aguilar, J.J. An overview of kinematic and calibration models using internal/external sensor or constraints to improve the behavior of spatial parallel mechanisms. *Sensors* **2010**, *10*, 10246–10297.
5. Zhong, X.-L.; Lewis, J.M.; Nagy, F.L.N. Autonomous robot calibration using a trigger probe. *Robot. Auton. Syst.* **1996**, *18*, 395–410.
6. Meng, Y.; Zhuang, H. Autonomous robot calibration using vision technology. *Robot. Comput. Integr. Manuf.* **2007**, *23*, 436–446.

7. Watanabe, A.; Sakakibara, S.; Ban, K.; Yamada, M.; Shen, G.; Arai, T. A Kinematic calibration method for industrial robots using autonomous visual measurement. *CIRP Ann. Manuf. Technol.* **2006**, *55*, 1–6.
8. Nahvi, A.; Hollerbach, J.M.; Hayward, V. Calibration of a Parallel Robot Using Multiple Kinematic Closed Loops. In Proceedings of the 1994 IEEE International Conference on Robotics and Automation, San Diego, CA, USA, 8–13 May 1994; pp. 407–412.
9. Aoyagi, S.; Kohama, A.; Nakata, Y.; Hayano, Y.; Suzuki, M. Improvement of Robot Accuracy by Calibrating Kinematic Model Using a Laser Tracking System-Compensation of Non-Geometric Errors Using Neural Networks and Selection of Optimal Measuring Points Using Genetic Algorithm. In Proceedings of the 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems, Taipei, Taiwan, 18–22 October 2010; pp. 5660–5665.
10. Ouyang, J. Ball array calibration on a coordinate measuring machine using a gage block. *Measurement* **1995**, *16*, 219–229.
11. Bai, S.; Teo, M.Y. Kinematic calibration and pose measurement of a medical parallel manipulator by optical position sensors. *J. Robot. Syst.* **2003**, *20*, 201–209.
12. Meng, G.; Tiemin, L.; Wensheng, Y. Calibration Method and Experiment of Stewart Platform Using a Laser Tracker. In Proceedings of IEEE International Conference on Systems, Man and Cybernetics, Washington, DC, USA, 5–8 October 2003; Volume 3, pp. 2797–2802.
13. Patel, A.J.; Ehmann, K.F. Calibration of a hexapod machine tool using a redundant leg. *Int J. Mach. Tools Manuf.* **2000**, *40*, 489–512.
14. Fazenda, N.; Lubrano, E.; Rossopoulos, S.; Clavel, R. Calibration of the 6 DOF High-Precision Flexure Parallel Robot “Sigma 6”. In Proceedings of the 5th Parallel Kinematics Seminar, Chemnitz, Germany, 22–26 April 2006; pp. 379–398.
15. Karlsson, B.; Brogårdh, T. A new calibration method for industrial robots. *Robotica* **2001**, *19*, 691–693.
16. Ren, X.; Feng, Z.; Su, C. A new calibration method for parallel kinematics machine tools using orientation constraint, *Int. J. Mach. Tools Manuf.* **2009**, *49*, 708–721.
17. Zhuang, H.; Masory, O.; Yan, J. Kinematic Calibration of a Stewart Platform Using Pose Measurements Obtained by a Single Theodolite. In Proceedings of International Conference on Intelligent Robots and Systems 95, “Human Robot Interaction and Cooperative Robots”, IEEE/RSJ, Pittsburgh, USA, 5–9 August 1995; Volume 2, pp. 329–334.
18. Großmann, K.; Kauschinger, B.; Szatmári, S. Kinematic calibration of a hexapod of simple design. *Prod. Eng. Springer Sept.* **2008**, *2*, 317–325.
19. Takeda, F.Y.; Shen, G.; Funabashi, H. A DBB-based kinematic calibration method for in-parallel actuated mechanisms using a Fourier series. *Trans. Am. Soc. Mech. Eng. J. Mech. Des.* **2004**, *126*, 856–865.
20. Chanal, H.; Duc, E.; Ray, P.; Hascoet, J. A new approach for the geometrical calibration of parallel kinematics machines tools based on the machining of a dedicated part. *Int. J. Mach. Tools Manuf.* **2007**, *47*, 1151–1163.
21. Canepa, G.; Hollerbach, J.M.; Boelen, A.J.M.A. Kinematic Calibration by Means of a Triaxial Accelerometer. In Proceedings of 1994 IEEE International Conference on Robotics and Automation, San Diego, CA, USA, 8–13 May 1994; Volume 4, pp. 2776–2782.

22. Motta, J.M.S.T.; de Carvalhob, G.C.; McMasterc, R.S. Robot calibration using a 3D vision-based measurement system with a single camera. *Robot. Comput. Integr. Manuf.* **2001**, *17*, 487–497.
23. Daney, D.; Andreff, N.; Chabert, G.; Papegay, Y. Interval method for calibration of parallel robots: Vision-based experiments. *Mech. Mach. Theory* **2006**, *41*, 929–944.
24. Andreff, N.; Martinet, P. Vision-based self-calibration and control of parallel kinematic mechanisms without proprioceptive sensing. *Intell. Serv. Robot.* **2009**, *2*, 71–80.
25. Renaud, P.; Andreff, N.; Lavest, J.; Dhome, M. Simplifying the kinematic calibration of parallel mechanisms using vision-based metrology. *IEEE Trans. Robot.* **2006**, *22*, 12–22.
26. Merlet, J.P. Interval analysis and reliability in robotics. *Int. J. Reliab. Saf.* **2009**, *3*, 104–130.
27. Dolinsky, J.U.; Jenkinson, I.D.; Colquhoun, G.J. Application of genetic programming to the calibration of industrial robots. *Comput. Ind.* **2007**, *58*, 255–264.
28. Amirat, M.Y.; Pontnau, J.; Artigue, F. A three-dimensional measurement system for robot applications. *J. Intell. Robot. Syst.* **1994**, *9*, 291–299.
29. Chiang, M.-H.; Lin, H.-T. Development of a 3D parallel mechanism robot arm with three vertical-axial pneumatic actuators combined with a stereo vision system. *Sensors* **2011**, *11*, 11476–11494.
30. Chiang, M.-H.; Lin, H.-T.; Hou, C.-L. Development of a stereo vision measurement system for a 3d three-axial pneumatic parallel mechanism robot arm. *Sensors* **2011**, *11*, 2257–2281.
31. Wang, D.; Bai, Y.; Zhao, J. Robot manipulator calibration using neural network and a camera-based measurement system. *Trans. Inst. Meas. Control* **2010**, *34*, 105–121.
32. Zhan, Q.; Wang, X. Hand–eye calibration and positioning for a robot drilling system. *Int. J. Adv. Manuf. Technol.* **2011**, *61*, 691–701.
33. Zhang, Z. A flexible new technique for camera calibration. *IEEE Trans. Pattern Anal. Mach. Intell.* **2002**, *22*, 1–21.
34. Clavel, R. Conception d’un robot parallele rapide a 4 degrés de liberté. Ph.D. Thesis, Ecole Polytechnique Fédérale De Lausanne: Lausanne, Switzerland, 1991.
35. Stamper, R. A Three Degree of Freedom Parallel Manipulator with Only Translational Degrees of Freedom. Ph.D. Thesis, University of Maryland: College Park, MD, USA, 1997.
36. Sebastian, J.M.; Traslosheros, A.; Angel, L.; Roberti, F.; Carelli, R. Parallel robot high speed object tracking. *Lect. Note. Comput. Sci.* **2007**, *4633*, 295–306.
37. Traslosheros, A.; Angel, L.; Vaca, R.; Sebastián, J.M.; Roberti, F. New Visual Servoing Control Strategies in Tracking Tasks Using a PKM. In *Mechatronic Systems Simulation Modeling and Control*; Milella, A., Di Paola, D., Cicirelli, G., Eds.; In-Tech: Lazinica, Vienna, Austria, 2010; pp. 117–146.
38. Ha, I.-C. Kinematic parameter calibration method for industrial robot manipulator using the relative position. *J. Mech. Sci. Technol.* **2008**, *22*, 1084–1090.
39. Besnard, S.; Khalil, W. Identifiable parameters for parallel robots kinematic calibration. *IEEE Int. Conf. Robot. Autom.* **2001**, *3*, 2859–2866.
40. Majarena, A.C.; Santolaria, J.; Samper, D.; Aguilar, J.J. Analysis an evaluation of objective function in kinematic calibration of parallel mechanisms. *Int. J. Adv. Manuf. Technol.* **2013**, *66*, 751–761.

41. European Committee for Standardization. Manipulating industrial robots. *Performance Criteria and Related Test Methods*; Geneva, Switzerland, 1998 (ISO 9283: 1998).
42. Chaumette, F.; Hutchinson, S. Visual servo control. Part I: Basic approaches. *IEEE Robot. Autom. Mag.* **2006**, *13*, 82–90.
43. Angel, L.; Traslosheros, A.; Sebastian, J.M.; Pari, L.; Carelli, R.; Roberti, F. Vision-based control of the RoboTenis system. *Lect. Notes Control Inf. Sci.* **2008**, *370*, 229–240.

Appendix

The following operation rules have been extracted from ISO 9283 (1998)

Position Repeatability

The repeatability of position expresses the dispersion of the reached positions after n repeated visits in the same direction to a prefixed position.

Let x_j, y_j, z_j be the coordinates of the j th reached position, and $\bar{x}, \bar{y}, \bar{z}$ the barycentre coordinates of the set of obtained points after repeating the same position n times, according to:

$$\bar{x} = \frac{1}{n} \sum_{j=1}^n x_j \quad ; \quad \bar{y} = \frac{1}{n} \sum_{j=1}^n y_j \quad ; \quad \bar{z} = \frac{1}{n} \sum_{j=1}^n z_j$$

The repeatability is defined as:

$$RP = \bar{l} + 3S$$

where:

$$\bar{l} = \frac{1}{n} \sum_{j=1}^n l_j \quad \text{with} \quad l_j = \sqrt{(x_j - \bar{x})^2 + (y_j - \bar{y})^2 + (z_j - \bar{z})^2}$$

and:

$$S = \sqrt{\frac{\sum_{j=1}^n (l_j - \bar{l})^2}{n-1}}$$

Positioning Distance Accuracy

Let P_{c1}, P_{c2} be the prefixed positions and P_{1j}, P_{2j} the reached positions after the j^{th} repetition, the positioning distance accuracy is defined as the difference between the prefixed positions and the mean of the difference among the reached.

Let x_{c1}, y_{c1}, z_{c1} and x_{c2}, y_{c2}, z_{c2} be the coordinates of the two prefixed positions and x_{1j}, y_{1j}, z_{1j} and x_{2j}, y_{2j}, z_{2j} the coordinates obtained in the j th reached position in a set of n repetitions, then the positioning distance accuracy is calculated by using:

$$AD = |\bar{D} - D_c|$$

where:

$$\bar{D} = \frac{1}{n} \sum_{j=1}^n D_j \quad \text{con} \quad D_j = |P_{1j} - P_{2j}| = \sqrt{(x_{1j} - x_{2j})^2 + (y_{1j} - y_{2j})^2 + (z_{1j} - z_{2j})^2}$$

and:

$$D_c = |P_{c1} - P_{c2}| = \sqrt{(x_{c1} - x_{c2})^2 + (y_{c1} - y_{c2})^2 + (z_{c1} - z_{c2})^2}$$

© 2013 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution license (<http://creativecommons.org/licenses/by/3.0/>).